Synthesis of Fault Diagnosable Logical Circuits

Masaru YAMAMOTO

Department of Management Engineering

(Received Aug. 18, 1972)

This paper describes a synthesis of fault diagnosable logical circuits. First, we introduce the combinational circuits suitable for the fault diagnosis, and present the algorithm for the fault diagnosis and the repairing method for these circuits. Second, we offer the optimal synthesis method for a given logical function by means of computer's aids. Last, we show that these circuits can be applied to sequential circuits.

1. Introduction

As reliability and availability are demanded of circuits, it has become an important problem to develope diagnosable logical circuits. [*1*], [*2*] Motivated by this problem, in this paper we first introduce a combinational circuit suitable for the fault diagnosis, and investigate the fault diagnosis and the repairing method for this circuit. Then, we show that these combinational circuits can be systematically diagnosed. Second, we investigate the optimal synthesis methed for a single logical output function by means of computer's. aids. We extend this synthesis method to the case of multi-output functions. We also show that these synthesis method can be applied in the case that these functions have some forbidden minterms, or "*don't care*". Last, it is shown that these circuits can be applied to sequential circuits, above all, to sequential circuits in the shift-register type.

2. Structure of the Circuit

In this paper, we intend to synthesize a given logical function by the circuit shown in *Fig. 1.* Then the circuit is divided into the two part :

(*I*) *The block part ;* In this part, we use the module shown in *Fig. 2.* as a building element, where every terminal y, terminal r, and terminal s of the modules in the i-th level are respectively connected to the bus lines y_i, r_i and s_i of the circuit. While the circuit is operating normally, every bus line r_i and s_i of the circuit are set such that $(r_i, s_i)=(1, 0)$. The constant value 0, 1, or an input variable can be applied to the external input terminals $\overset{k}{c}_j$ (where, $k=0, 1, \cdots\cdots, 2^{n-s}-1$) of the j-th block. Eor brevity, set $(\overset{0}{c}_j, \overset{1}{c}_j, \cdots\cdots, c_j^{2^{n-s}-1})=\mathbf{a}_j$. We apply the input variables $x_{s+1}, x_{s+2}, \cdots\cdots, x_{n-1}$ to the bus lines $y_1, y_2, \cdots\cdots, y_{n-s-1}$ respectively. Then, two output functions of the block j are as follows

$$f_0^{(j)}(x_{s+1}, x_{s+2}, \cdot\cdot, x_n)=\sum_{i=0}^{2^{n-s-1}-1} c_j^i\, m_i, \tag{1}$$

$$f_1^{(j)}(x_{s+1}, x_{s+2}, \cdot\cdot, x_n)=\sum_{i=2^{n-s-1}}^{2^{n-s}-1} c_j^i\, m_i. \tag{2}$$

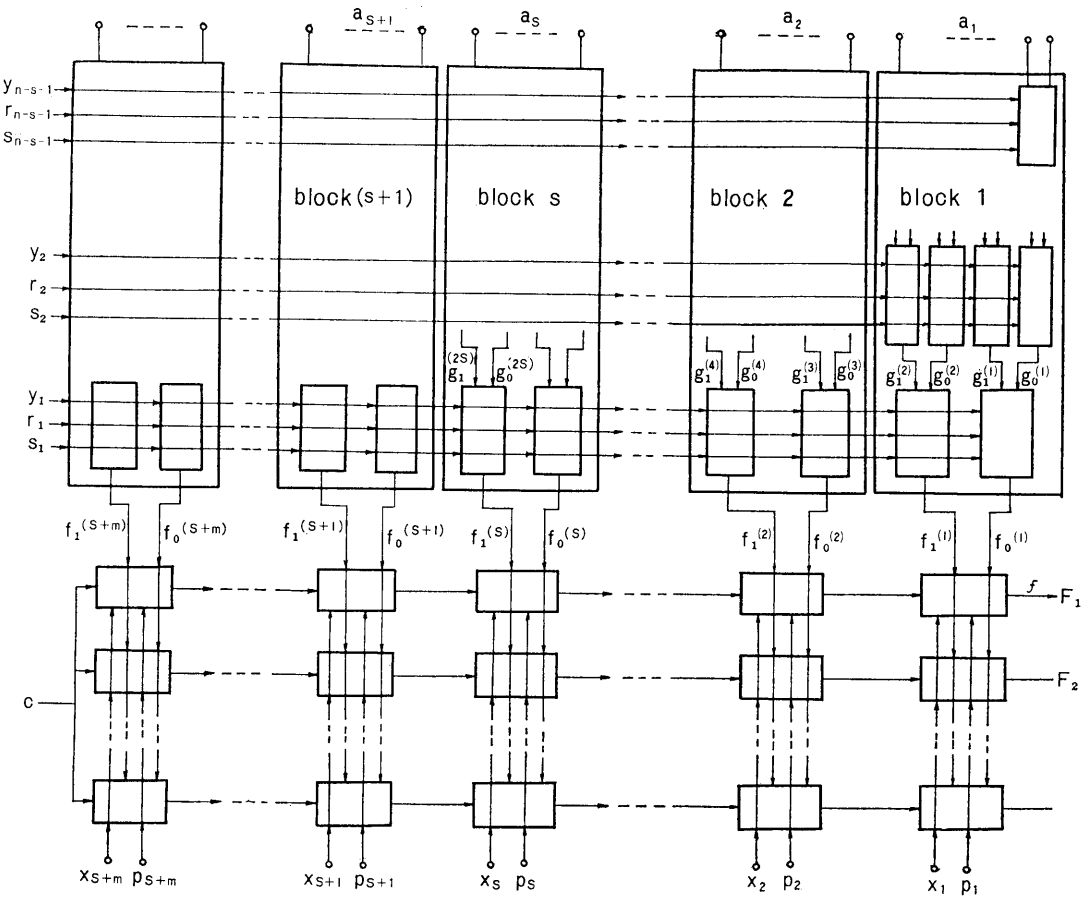
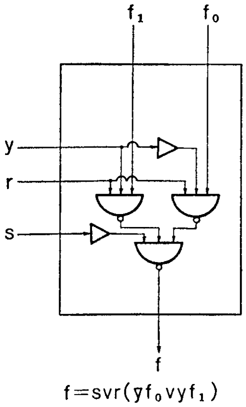


Fig. 1 Combinational circuit realization

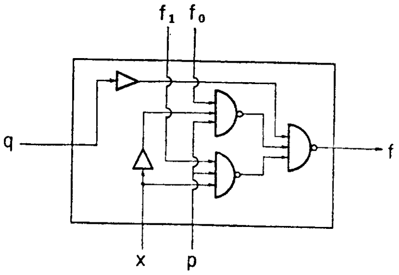
where, m_i is a minterm of variables $x_{s+1}, x_{s+2}, \cdots\cdots, x_{n-1}$, and each c_j^i takes the value *0, 1,* or x_n $(\bar{x}_n)$.

(*II*) *The collector part* ; We use the module shown in *Fig. 3.* as a building element. This building element is essentialy same with that of *Fig. 2.* The output function $f(x_1, x_2, \cdots\cdots, x_n)$ from a terminal F_1 is represented as follows



$f = s \vee r(\bar{y} f_0 \vee y f_1)$

Fig. 2 Module in the block part of the circuit



$f = q \vee p(\bar{x} f_0 \vee x f_1)$

Fig. 3 Module in the collector part of the circuti

$$f(x_1, x_2, \cdots\cdots, x_n) = p_1(\bar{x}_1 f_0^{(1)} \vee x_1 f_1^{(1)}) \vee p_2(\bar{x}_2 f_0^{(2)} \vee x_2 f_1^{(2)}) \vee \cdots\cdots\cdots \vee p_s(\bar{x}_s f_0^{(s)} \vee x_s f_1^{(s)}) \vee \cdots\cdots\cdots \vee C \quad (3)$$

where each function $f_0^{(j)}$ ($f_1^{(j)}$) is a logical function of variables $x_{s+1}, x_{s+2}, \cdots\cdots, x_n$ shown in *Eqs* (*1*) and (*2*). Note that the block part $(s+1)$, $(s+2)$, $\cdots\cdots$, $(s+m)$ are the spare blocks for the repair in the case when the failures occur in the circuit. While the circuit is operating normally, we apply *0* to the terminals p_i $(i=s+1, s+2, \cdots\cdots, s+m)$ so that these spare blocks should not affect the circuit. We also apply $p_j=1$ $(j=1, 2, \cdots\cdots, s)$ and $C=0$. Then the output function f of *Eq*. (*3*) is given by

$$f(x_1, x_2, \cdots, x_n) = (\bar{x}_1 f_0^{(1)} \vee x_1 f_1^{(1)}) \vee (\bar{x}_2 f_0^{(2)} \vee x_2 f_1^{(2)}) \vee \cdots\cdots\cdots \vee (\bar{x}_s f_0^{(S)} \vee x_s f_1^{(S)}) . \quad (4)$$

The circuit of *Fig. 1* posseses the following properties.

<property. 1> The output function f of the circuit can be set as $f=\bar{x}_j f_0^{(j)} \vee x_j f_1^{(j)}$, provided that $C=0$, $p_j=1$, $p_r=0(r=1, 2, \cdots\cdots, j-1, j+1, \cdots\cdots, s+m)$.

<property. 2> If the failures are detected in the block j, we can exchange the faulty block for a spare block $(s+1)$, provided that $p_{s+1}=1$, $p_j=0$, $x_{s+1}=x_j$, $\mathbf{a}_{s+1}=\mathbf{a}_j$.

<property. 3> If the failures occur in the collector part, we can use the next output terminal F_2.

<property. 4> In order to set the output lines of all modules in the i-th level of the block part to *0*, we only set $(r_i, s_i)=(0, 0)$. Similarly the lines can be set to *1*, provided that $(r_i, s_i)=(\times, 1)$.

3. Fault Diagnosis of the Circuit

In this section we investigate the fault diagnosis of the combinational circuit of *Fig. 1*. The type of the fault is assumed to be either stuck-at-*1* (for brevity, ***s-a-1***), or stuck-at-*0* (***s-a-0***) at the lines connecting modules. The circuit may have any number of faults. By means of the *properties 1～4*, we can obtain the procedure for the fault diagnosis and the repairing method for the circuit.

3. 1 Fault diagnosis for the collector part

First of all, we diagnose the collector part by the following method. If the failure is detected, we can repair the collector part by *property. 3*. In order to diagnose the collector part, we must make an activated path from the terminal C to the output terminal F_1. Therefore we apply the next two testing inputs (A_1, A_2).

$$\left.\begin{array}{l} A_1:\ C=1 \\ A_2:\ C=0 \end{array}\right\} \textit{in common, } p_i=0\ (i=1, 2, \cdots, s+m).$$

Then, if we gain the output pattern (*1, 0*) at the output terminal F_1, we can conclude that the collector part operates normally. Otherwise the collector part is proved to be faulty.

3. 2 Fault diagnosis for the block part

After repairing the collector part, we diagnose the block part by the following procedure. If the faulty block is located, then we can repair the block for a new one by utilizing *property. 2.*

First, we diagnose the lines $\boldsymbol{L}_1=\{f_0^{(1)}, f_1^{(1)}, \cdots\cdots, f_0^{(s)}, f_1^{(s)}\}$ by applying the next testing inputs ($\boldsymbol{T}_1$, $\boldsymbol{T}_2$).

$$\left.\begin{aligned} \boldsymbol{T}_1 &: (x_1, x_2, \cdots\cdots, x_s)=(0, 0, \cdots\cdots, 0) \\ \boldsymbol{T}_2 &: (x_1, x_2, \cdots\cdots, x_s)=(1, 1, \cdots\cdots, 1) \end{aligned}\right\} \begin{aligned} &\textit{in common}, C=0, (r_1, s_1)=(0, 0), p_j=1 \ (j=1, 2, \\ &\cdots\cdots, s), p_i=0 \ (i=s+1, \cdots\cdots, s+m). \end{aligned}$$

When the testing input $\boldsymbol{T}_1$ is applied to the circuit, the output function $\boldsymbol{f}$ of *Eq.* (*4*) is given as follows

$$f= f_0^{(1)} \vee f_0^{(2)} \vee \cdots \vee f_0^{(j)} \vee \cdots \vee f_0^{(s)} . \tag{5}$$

Therefore, if there exists no the $\boldsymbol{s\text{-}a\text{-}1}$ faulty line in $f_0^{(1)}, f_0^{(2)}, \cdots\cdots, f_0^{(s)}$, we gain $\boldsymbol{f=0}$ at the F_1 by setting $(r_1, s_1)=(0, 0)$. However if there exists the $\boldsymbol{s\text{-}a\text{-}1}$ faulty line in them, we gain $\boldsymbol{f=1}$. Similarly, the output function $\boldsymbol{f}$ of *Eq.* (*4*) is shown in *Eq.* (*6*) in the case of $\boldsymbol{T}_2$'s applying

$$f= f_1^{(1)} \vee f_1^{(2)} \vee \cdots \vee f_1^{(j)} \vee \cdots \vee f_1^{(s)} . \tag{6}$$

The same discussion can be done about $\boldsymbol{T}_2$. As a result, we can conclude: (**i**) We gain the output (*0, 0*) at the F_1, when the testing inputs ($\boldsymbol{T}_1$, $\boldsymbol{T}_2$) are applied. In this case, there exists no the $\boldsymbol{s\text{-}a\text{-}1}$ faulty line in $\boldsymbol{L}_1$. (**ii**) In the case of (*0, 1*), there exist the $\boldsymbol{s\text{-}a\text{-}1}$ faulty lines in $f_1^{(1)}, f_1^{(2)}, \cdots\cdots$, *or* $f_1^{(s)}$. In order to locate the faulty lines, we apply the next s 's testing inputs.

$$\left.\begin{aligned} \boldsymbol{H}_1 &: x_1=1, (p_1, p_2, \cdots\cdots, p_s)=(1, 0, \cdots\cdots, 0) \\ \boldsymbol{H}_2 &: x_2=1, (p_1, p_2, \cdots\cdots, p_s)=(0, 1, \cdots\cdots, 0) \\ &\vdots \\ \boldsymbol{H}_s &: x_s=1, (p_1, p_2, \cdots\cdots, p_s)=(0, 0, \cdots\cdots, 1) \end{aligned}\right\} \begin{aligned} &\textit{in common}, C=0, (r_1, s_1)=(0, 0), \\ &p_i=0 \ (i=s+1, \cdots\cdots, s+m) \end{aligned}$$

If we gain the output *1* at the F_1, by applying $\boldsymbol{H}_2$ and $\boldsymbol{H}_s$. we can conclude that lines $f_1^{(2)}$ and $f_1^{(s)}$ are faulty. (**iii**) In the case of (*1, 0*), there exist the $\boldsymbol{s\text{-}a\text{-}1}$ faulty lines in $f_0^{(1)}, f_0^{(2)}, \cdots\cdots, f_0^{(s)}$. We also can locate the faulty lines by applying the next testing inputs.

$$\begin{aligned} \boldsymbol{J}_1 &: x_1=0, (p_1, p_2, \cdots\cdots, p_s)=(1, 0, \cdots\cdots, 0) \\ \boldsymbol{J}_2 &: x_2=0, (p_1, p_2, \cdots\cdots, p_s)=(0, 1, \cdots\cdots, 0) \\ &\vdots \\ \boldsymbol{J}_s &: x_s=0, (p_1, p_2, \cdots\cdots, p_s)=(0, 0, \cdots\cdots, 1) \end{aligned} \quad \begin{aligned} &\textit{in common}, C=0, (r_1, s_1)=(0, 0), \\ &p_i=0 \ (i=s+1, \cdots\cdots, s+m). \end{aligned}$$

(**iv**) In the case of (*1, 1*), we can locate the $\boldsymbol{s\text{-}a\text{-}1}$ faulty lines in $\boldsymbol{L}_1$ by applying the $\boldsymbol{H}_1$, $\boldsymbol{H}_2$, $\cdots\cdots$, $\boldsymbol{H}_s$, $\boldsymbol{J}_1$, $\boldsymbol{J}_2$, $\cdots\cdots$, and $\boldsymbol{J}_s$.

Next we diagnose the $\boldsymbol{s\text{-}a\text{-}0}$ faulty lines in $\boldsymbol{L}_1$. So that we apply the following *2s* 's testing inputs to the circuit.

$$\left.\begin{aligned} \boldsymbol{D}_1(\boldsymbol{E}_1) &: x_1=1 \ (0), (p_1, p_2, \cdots\cdots, p_s)=(1, 0, \cdots\cdots, 0) \\ \boldsymbol{D}_2(\boldsymbol{E}_2) &: x_2=1 \ (0), (p_1, p_2, \cdots\cdots, p_s)=(0, 1, \cdots\cdots, 0) \\ &\vdots \\ \boldsymbol{D}_s(\boldsymbol{E}_s) &: x_s=1 \ (0), (p_1, p_2, \cdots\cdots, p_s)=(0, 0, \cdots\cdots, 1) \end{aligned}\right\} \begin{aligned} &\textit{in common}, C=0, s_1=1, p_i=0 \\ &(i=s+1, \cdots\cdots \ s+m). \end{aligned}$$

Suppose we apply testing input $\boldsymbol{D}_j$ to the circuit. The output function at the F_1 can be expressed as follows

$$f = f_1^{(j)} \tag{7}$$

Hence, we gain the output *1* at the F_1, if there exists no the ***s-a-O*** faulty in the line $f_1^{(j)}$. Otherwise we gain the output *0*. As easily known from the above considerations, we can locate the ***s-a-O*** faulty lines in L_1, by observing the output at the F_1.

In the same way we can continue to diagnose the lines $L_2=\{g_0^{(1)}, g_1^{(1)}, g_0^{(2)}, g_1^{(2)}, \cdots\cdots, g_0^{(2s)}, g_1^{(2s)}\}$, L_3, ……, L_{n-s-1} successively. The details are to be referred to the authors' note[3].

4. Synthesis of a Logical Function

In what follows we investigate the synthesis of a logical function using the circuit shown in *Fig. 1*.

4. 1 Definitions

In preparation for later discussions, we give necessary definitions in the following.

<Definition. 1> G is a subset of N. $\overline{G}$ means that $\overline{G}=N\backslash G$, where $N=\{x_1, x_2, \cdots\cdots, x_n\}$.

<Definition. 2> $|G|$ denotes the number of variables in G.

<Definition. 3> $G=\{x_{i_1}, x_{i_2}, \cdots\cdots, x_{i_s}\}$ is called a set of independent variables for the function f_n iff the function can be expanded with respect to G as follows

$$f_n=(\overline{x}_{i_1}f_0^{(1)} \vee x_{i_1}f_1^{(1)}) \vee (\overline{x}_{i_2}f_0^{(2)} \vee x_{i_2}f_1^{(2)}) \vee \cdots\cdots \vee (\overline{x}_{i_j}f_0^{(j)} \vee x_{i_j}f_1^{(j)}) \vee \cdots \vee (\overline{x}_{i_s}f_0^{(s)} \vee x_{i_s}f_1^{(s)}) \tag{8}$$

where, each $f_0^{(j)}$ (or, $f_1^{(j)}$) is a function of variables $x_{i_{s+1}}, x_{i_{s+2}}, \cdots\cdots, x_{i_n}$.

<Definition. 4> $\overline{G}$ (a_i) is a function which means a vector composed of elements in a_i whose correspoding variables exist in $\overline{G}$, where a_i is a row vector with n-dimension, corresponding to a minterm of f_n.

<Definition. 5> For a given function f_n, G is called a maximum set of independent variables for f_n, iff there exists no set of independent variables G' which satisfies $G' \supsetneqq G$.

<Definition. 6> For a given function f_n, we define the largest width w of f_n as follows

$$w = \max_j |G_j| \tag{9}$$

where, G_j is a maximum set of indepenent variables for f_n.

4. 2 A few comments on the optimal circuits

Assume that an n-variables logical function f_n can be expanded with respect to $G=\{x_{i_1}, x_{i_2}, \cdots\cdots x_{i_w}\}$ as follows

$$f_n=(\overline{x}_{i_1}f_0^{(1)} \vee x_{i_1}f_1^{(1)}) \vee (\overline{x}_{i_2}f_0^{(2)} \vee x_{i_2}f_0^{(2)}) \vee \cdots \vee (\overline{x}_{i_w}f_0^{(w)} \vee x_{i_w}f_1^{(w)}) \cdot \tag{10}$$

where, G gives the largest width w of f_n. Hence the function f_n can be synthesized as shown in *Fig. 4*. Then, the number M of modules needed for synthesizing the logical function f_n, the number B of modules contained in a block of the circuit, and the number T of external input-output terminals of the circuit, are as follows

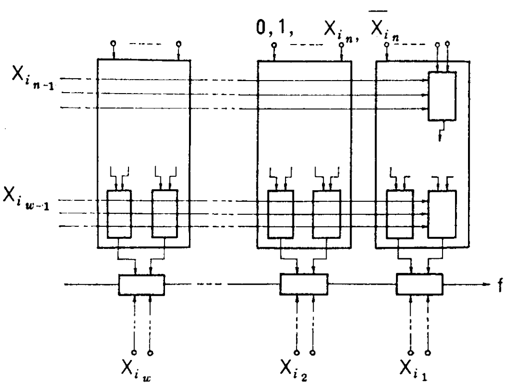


Fig. 4 Circuit realization with G$=\{x_{i_1}, x_{i_2}, \cdots\cdots, x_{i_w}\}$

$$M = w\,(2^{n-w}-1)\ , \qquad (11)$$

$$B = 2^{n-w} - \ 2\ , \qquad (12)$$

$$T = w\cdot 2^{n-w} + 3n - w - 1\,. \qquad (13)$$

M, ***B*** and ***T*** decrease as w increases. Therefore we obtain the next theorems and lemma.

[***Theorem. 1***] *G is a set of independent variables for* f_n *iff for any* α_i *satisfying* $f_n(\alpha_i)=1$, *there exists a variable* x_i, $x_i \in G$, *and the next condition 1 or 2 is satisfied.*

Condition .1 ; *There exist* $2^{|G|-1}$ *'s vectors* β_j *which satisfy* $f_n(\beta_j)=1$, $\overline{G}(\alpha_i)=\overline{G}(\beta_j)$, $x_i(\beta_j) = (1)$.

Condition. 2 ; *There exist* $2^{|G|-1}$ *'s vectors* β_j *which satisfy* $f_n(\beta_j)=1$, $\overline{G}(\alpha_i)=\overline{G}(\beta_j)$, $x_i(\beta_j) = (0)$.

We can obtain the next lemma from the above theorem.

[***Lemma. 1***] *In the ON-array of* f_n[4], *if there exists a vector* α_i *which does not satisfy* $\overline{G}(\alpha_i)=\overline{G}(\beta_{i_1})= \cdots\cdots = \overline{G}(\beta_{i_k})$, *where* $k \geq 2^{|G|-1}$, $\beta_{i_j} \in$ *ON-array, then* ***G*** *is not a set of independent variables for* f_n.

[***Theorem. 2***] *We can obtain the optimal circuit by seeking the set of independent variables* ***G*** *for* f_n, *where,* $|G|$ *is the largest width of* f_n.

Based on these theorems and lemma, we will consider a method of synthesizing an optimal circuit in the next section.

4. 3 *The algorithm for the optimal circuit synthesis*

Given a logical function f_n, we synthesize the optimal circuit by the following algorithm.

step. 1) An n-variables function f_n is given in the form of *ON-array.*

step. 2) As an initial condition, set $i=n$.

step. 3) Seek a subset of variables G in order, and go to *step. 4.* where, $|G|=i$. If all G are checked, go to *step. 6*).

step. 4) According to *Lemma. 1,* examine whether this ***G*** satisfies the necessary condition for a set of independent variables for f_n or not. If the condition is satisfied, then go to *step. 5*)' otherwise go to *step. 3*).

step. 5) According to *Theorem. 1*, examine the sufficient condition of ***G***. If it is satisfied, go to *step. 7*), otherwise go to *step. 3*).

step. 6) Set $\boldsymbol{i=i-1}$, then go to *step. 3*).

step. 7) Synthesize the circuit by expanding $\boldsymbol{f_n}$ with respect to the ***G***.

4. 4 *Example*

Now we explain briefly the synthesis method mentioned above with the use of the next example.

〔***Example. 1***〕 Given a logical function f expressed as the following form

$$f = x_1\bar{x}_3x_5x_8 \vee x_2\bar{x}_3\bar{x}_5\bar{x}_8 \vee x_3\bar{x}_4\bar{x}_5x_8 \vee \bar{x}_3x_5x_6x_8 \vee x_3x_5\bar{x}_6x_8 \vee \bar{x}_3\bar{x}_5x_7x_8 \tag{14}$$

By applying to ***f*** the synthesis method of ***4.3***, we can get the set of independent variables ***G*** in *Eq.(15)* which give the largest width for ***f***.

$$G = \{x_1, x_2, x_4, x_6, x_7\} \tag{15}$$

Therefore we can expand the logical function f with respect to ***G*** and synthesize the optimal circuit for ***f*** as shown in *Fig. 5*.

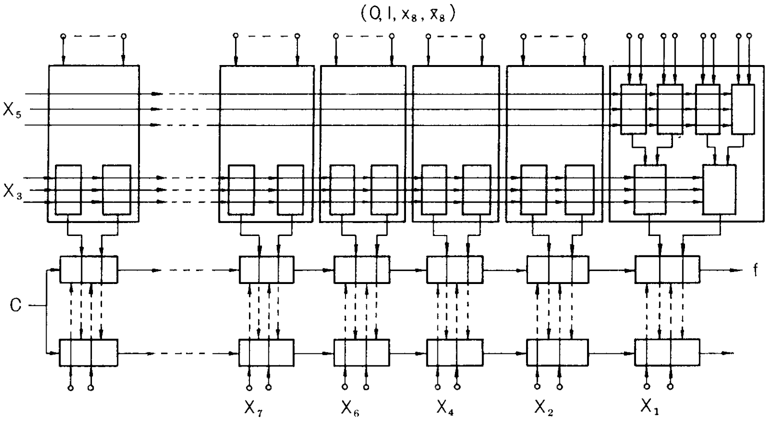


Fig. 5 Circuit realization for Eq. (14)

4. 5 *Synthesis of a logical function with "don't care"*

We give the synthesis method for the optimal circuit in ***4.3***. In this section we also show this method can be applied in the case that the logical function $\boldsymbol{f_n}$ has *"don't care"*, with minor modifications.

For a logical function $\boldsymbol{f_n}$ with *"don't care"*, the *Theorem. 1* and *Lemma. 1* can be rewritten as follows.

〔***Theorem. 3***〕 *G is a set of independent variables for* f_n *iff, for any* $\boldsymbol{a}_i \in$ *ON-array, there exists a variable* x_i, $x_i \in G$ *and the next condition 1 or 2 is satisfied.*

Condition. 1 ; *There exist* $2^{|G|-1}$ *'s vectors* $\boldsymbol{\beta}_j$, $\boldsymbol{\beta}_j \in$ *ON-array* ∪ *DC-array,* which satisfy* $\bar{G}(\boldsymbol{a}_i) = \bar{G}(\boldsymbol{\beta}_j)$, $x_i(\boldsymbol{\beta}_j) = (1)$.

* *DC-array* denotes the set of minterms of *"don't care"* .

Condition. 2 *; There exist $2^{|G|-1}$'s vectors $\boldsymbol{\beta}_j$, $\boldsymbol{\beta}_j \in$ ON-array $\cup$ DC-array, which satisfy $\overline{G}(\boldsymbol{a}_i)=\overline{G}(\boldsymbol{\beta}_j)$, $x_i(\boldsymbol{\beta}_j)=(0)$.*

〔Lemma. 2〕 In the ON-array of $\boldsymbol{f}_n$ if there exists a vector $\boldsymbol{a}_i$ which does not satisfy $\overline{\boldsymbol{G}}(\boldsymbol{a}_i)=\overline{\boldsymbol{G}}(\boldsymbol{\beta}_{i_1})=\overline{\boldsymbol{G}}(\boldsymbol{\beta}_{i_2})=\cdots\cdots=\overline{\boldsymbol{G}}(\boldsymbol{\beta}_{i_k})$, where, $k \geq 2^{|G|-1}$, $\boldsymbol{\beta}_{i_j} \in$ ON-array $\cup$ DC-array, then G is not a set of independent variables for $\boldsymbol{f}_n$.

Therefore we can utilize the synthesis method in ***4.3*** with some modifications.

5. Synthesis of Multi-Output Functions

5. 1 The synthesis method

In this section we discuss abut the synthesis method for multi-output functions. In this case the circuit configuration in *Fig. 1* must be modified a little. This modified circuit is shown in *Fig. 6*, and also verified to be fault diagnosable circuit.

Next we investigate the synthesis method for the optimal circuit for multi-output functions. Assume that functions $\boldsymbol{f}_1, \boldsymbol{f}_2, \cdots, \boldsymbol{f}_m$ are given. If these functions can be expanded as follows

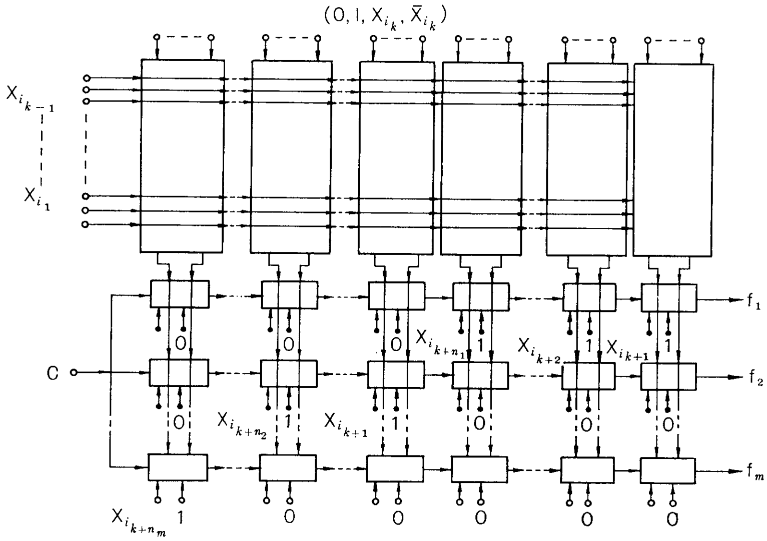


Fig. 6 Circuit realization for multi-output functions

$$\left.\begin{aligned}
f_1 &= g_1^1(x_{i_{K+1}}, K) \vee g_1^2(x_{i_{K+2}}, K) \vee \cdots \vee g_1^{n_1}(x_{i_{K+n_1}}, K),\\
f_2 &= g_2^1(x_{i_{K+1}}, K) \vee g_2^2(x_{i_{K+2}}, K) \vee \cdots \vee g_2^{n_2}(x_{i_{K+n_2}}, K),\\
&\vdots\\
f_m &= g_m^1(x_{i_{K+1}}, K) \vee g_m^2(x_{i_{K+2}}, K) \vee \cdots \vee g_m^{n_m}(x_{i_{K+n_m}}, K).\\
&\text{where, } K = \{x_{i_1}, x_{i_2}, \ldots, x_{i_K}\}
\end{aligned}\right\} \quad (16)$$

Then, we can synthesize these multi-output functions by the circuit shown in *Fig. 6*. As easily known, it is necessary, for obtaining the optimal circuit for multi-output functions, to seek the $\boldsymbol{K}$ whose $|\boldsymbol{K}|$ is the least number. We call the $\boldsymbol{K}$ as $\boldsymbol{K}_{min}$. By using *Definitions 1* and *3*, $\boldsymbol{K}_{min}$ can be represented as follows

$$K_{min} = \min_{j} \overline{\bigcup_{i=1\sim m} G_i^j} \qquad (17)$$

where, G_i^j is a set of independent variables for f_i. Hence, we can obtain $\boldsymbol{K}_{min}$ by applying the algorithm of ***4.3*** and *Eq. (17)*. We also research the synthesis method for multi-output functions, and have programmed it.

5. 2 Example

As an example we synthesize an optimal circuit for *Eq. (18)*. The circuit obtained by the programming is shown in *Fig. 7*, where, $\boldsymbol{K}_{min}=\{x_2, x_4, x_6, x_7\}$, $|\boldsymbol{K}_{min}|=4$.

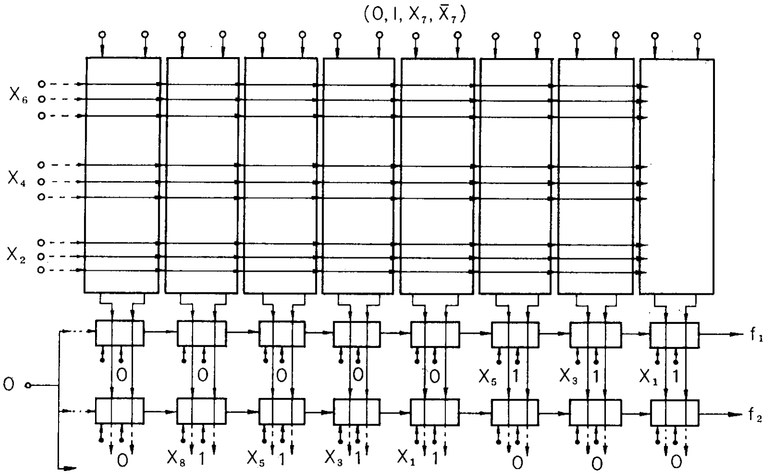


Fig. 7 Circuit realization for logical functions f_1 and f_2 of Eq. (18)

$$\left.\begin{aligned} f_1 = {} & \bar{x}_1\bar{x}_2\bar{x}_4\bar{x}_6 \vee x_2x_3\bar{x}_4\bar{x}_6 \vee \bar{x}_2x_4\bar{x}_6 \vee x_1x_7 \\ & \vee x_2x_4x_6\bar{x}_7 \vee \bar{x}_2x_3\bar{x}_4x_6 \vee x_2\bar{x}_4x_5x_6 \quad , \\ f_2 = {} & \bar{x}_2x_4\bar{x}_6\bar{x}_7 \vee x_1\bar{x}_4x_6\bar{x}_7 \vee x_4x_6\bar{x}_8 \\ & \vee \bar{x}_4\bar{x}_5x_6x_7 \vee x_4x_7 \end{aligned}\right\} \quad (18)$$

6. Extension to the Sequential Circuits

Here we show that these circuits can be applied to sequential circuits. Concerning the diagnosable sequential circuits, a number of authors have discussed,[5], [6] In this section, we consider a synthesis of sequential circuit in the shift-register type[7],[8] shown in *Fig. 8*, and show that these sequential circuits can be easily diagnosed. In the circuit of *Fig. 8*, the transient function $\boldsymbol{f}$ can be expressed as follows

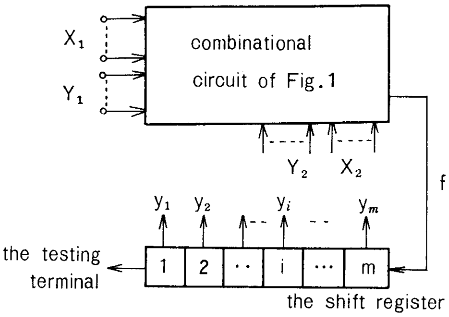


Fig. 8 The block diagram of the sequential circuit in the shift-register type

where, X_1, X_2 ; the set of external input variables
$X_1 \cup X_2 = \{x_1, x_2, \cdots\cdots, x_n\}$, $X_1 \cap X_2 = \phi$
Y_1, Y_2 ; the set of variables from the shift register
$Y_1 \cup Y_2 = \{y_1, y_2, \cdots\cdots, y_m\}$, $Y_1 \cap Y_2 = \phi$

$$f = \varphi_1^1(X_1, Y_1, x_{i_1}) \vee \varphi_1^2(X_1, Y_1, x_{i_2}) \vee \cdots \vee \varphi_1^k(X_1, Y_1, x_{i_k}) \\ \vee \varphi_1^{k+1}(X_1, Y_1, y_{i_1}) \vee \cdots \vee \varphi_1^{k+\ell}(X_1, Y_1, y_{i_\ell}) \ . \qquad (19)$$

where, $X_2 = \{x_{i_1}, x_{i_2}, \cdots\cdots, x_{i_k}\}$,
$Y_2 = \{y_{i_1}, y_{i_2}, \cdots\cdots, y_{i_l}\}$.

Therefore, it is necessary, for obtaining the optimal sequential circuit in the shift-register type, to seek the set of independent variables $X_2 \cup Y_2$ for f, where $|X_2| + |Y_2|$ is the largest number. However in synthesizing the optimal sequential circuit in the shift-register type, we must give careful considerations to the partition of equivalent states, states assignments, and so on.

7. Conclusion

In this paper we have considered the fault diagnosable logical circuits. Then we have shown that it is necessary, for obtaining the optimal circuit, to seek the set of independent variables **G** which give the largest width *w*. So that we give the algorithm for synthesizing the optimal circuit, and have programmed it. We have also shown these synthesis method can be applied in the case of function with "*don't care*" and multi-output functions. Furthermore we have extended these fault diagnosable circuits to the sequential circuits. As an example we have presented a sequential circuit in the shift-register type.

8. Acknowledgement

Authors would like to thank *pro. H. Ozaki* and *associate Pro. K. Kinoshita* of the *Osaka University* for their supports and encouragements. Thanks are also due to *Mr. H. Ueda* (*Suwa Seiko co*) for his helps in programming the synthesis methods.

Reference

〔1〕 M. Yamamoto, K. Kinoshita ; Jour. **I. E. C. E.,** Japan, **54-C, 9,** p. 804 (1971-09). (in Japanese)

〔2〕 S. S. Yau, C. K. Tang ; **IEEE** Trans., **C-19** (Feb. 1970).

〔3〕 M. Yamamoto, K. Kinoshita ; Jour. **I. E. C. E.,** Japan, **55-D, 1,** p. 31 (1972-01). (in Japanese)

〔4〕 J. P. Roth, R. M. Karp ; **IBM** Jour., p. 227 (Apr. 1962).

〔5〕 Z. Kohavi, P. Lavallee; **IEEE** Trans., **EC-16,** p. 473 (1969).

〔6〕 S. Murakami, K. Kinoshita, H. Ozaki ; Jour. **I. E. C. E.,** Japan' **51,** p. 455 (1968). (in Japanese)

〔7〕 T. Kawada, K˙ Kinoshita, H. Ozaki ; Jour. **I. E. C. E.,** Japan, **52-C, 7,** p. 378 (1969-07). (in Japanese)

〔8〕 R. L. Martin ; IEEE conf. on Switching and Automata Theory, p. 240 (Oct. 1967).

〔9〕 M. Yamamoto ; Paper of Conv. **I. E. C. E.,** Japan, **Sl-4,** (1972-04), (in Japanese)

〔10〕 M. Yamamoto ; Jour. **I. E. C. E.,** Japan, **55-D, 9** (1972-09). (in Japanese)